EMPLOYEE 20
RECORDS
1,....m
(MODEL)

30

```
{
  Query employee DB (model)
  Foreach employee in queryResult
    Create "Detail" button
    Create employee-name label
  end
}
```

FWApplication (Controller & View-Generation Logic)

10

FILE EDIT VIEW HELP

EMPLOYEE | RAY | SEARCH

10-3
10-1
10-2

(INITIAL VIEW)

40

FILE EDIT VIEW HELP

EMPLOYEE | RAY | SEARCH

DETAILS RAYFIELD, JAMES T.
DETAILS RAY, DAVID
DETAILS RAYMOND, DOUG

40-4
40-5
40-6
40-1
40-2
40-3

(NEW VIEW)

FIG. 1

```
public interface ViewAPI
{
11-1 →    public ViewElement createViewElement(ViewType viewType, String name);

11-2 →    public void removeViewElement(String name);

11-3 →    public ViewElement getViewElement(String name);
}


public interface ViewElement
{
11-4 →    public ViewProperty getViewProperty(String name);

11-5 →    public void setViewProperty(String name, PropertyValue value);

11-6 →    public void registerController
              (Controller controller, ControllerArguments args, UserInteraction event);
}
```

~ 11

FIG. 2

```
public interface ModelAPI
{
12-1 →    public ModelHome locateHome (String homeName);
}

public interface ModelHome
{
12-2 →    public ModelElement createModelElement (ModelElementKey key);

12-3 →    public void removeModelElement (ModelElementKey key);

12-4 →    public ModelElementSet Query (QueryCriteria criteria);
}

public interface ModelElement
{
12-5 →    public ModelElementKey getKey();
}
```

~ 12

FIG. 3

YOR920000470US1

30

```
// Controller

30-0 ──→   public void ProcessSearchRequest
           {
               // access the view components
30-10 ──→      TextField    inputName = ViewAPI.getViewElement(inputKey);
30-20 ──→      String       employeeName = inputName.getProperty(data);

               // access the model
30-30 ──→      EmployeeHome   employeeHome = ModelAPI.locateHome(employees);
               // query the model
30-40 ──→      Employee[]   results =
                   employeeHome.Query (new QueryCriteria (findByName, employeeName));

               // generate the next view, and return to client
30-50 ──→      setView(BuildSearchResultsView (name, results));
           }


           // View Generation code

30-60 ──→   public void CreateInitialView ()
           {
               // create the top-level frame
30-70 ──→      Frame     frame = ViewAPI.createViewElement (frame, frameKey);
30-80 ──→      TextField  inputName = ViewAPI.createViewElement (textfield, inputKey);

               // associate a controller with user-input into the textfield
30-90 ──→      inputName.registerController (ProcessSearchRequest, null, TextField.CHANGE);
           }
```
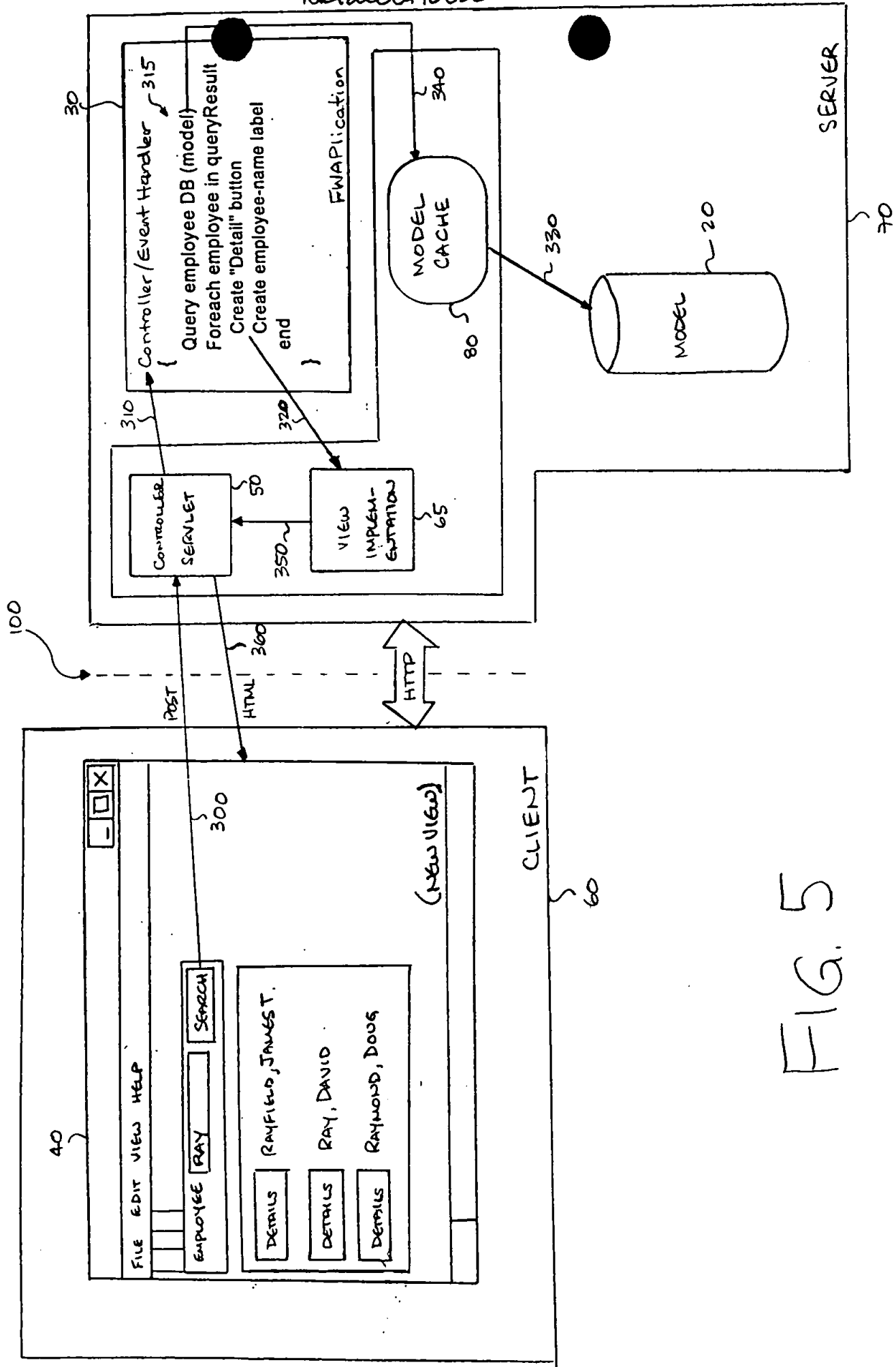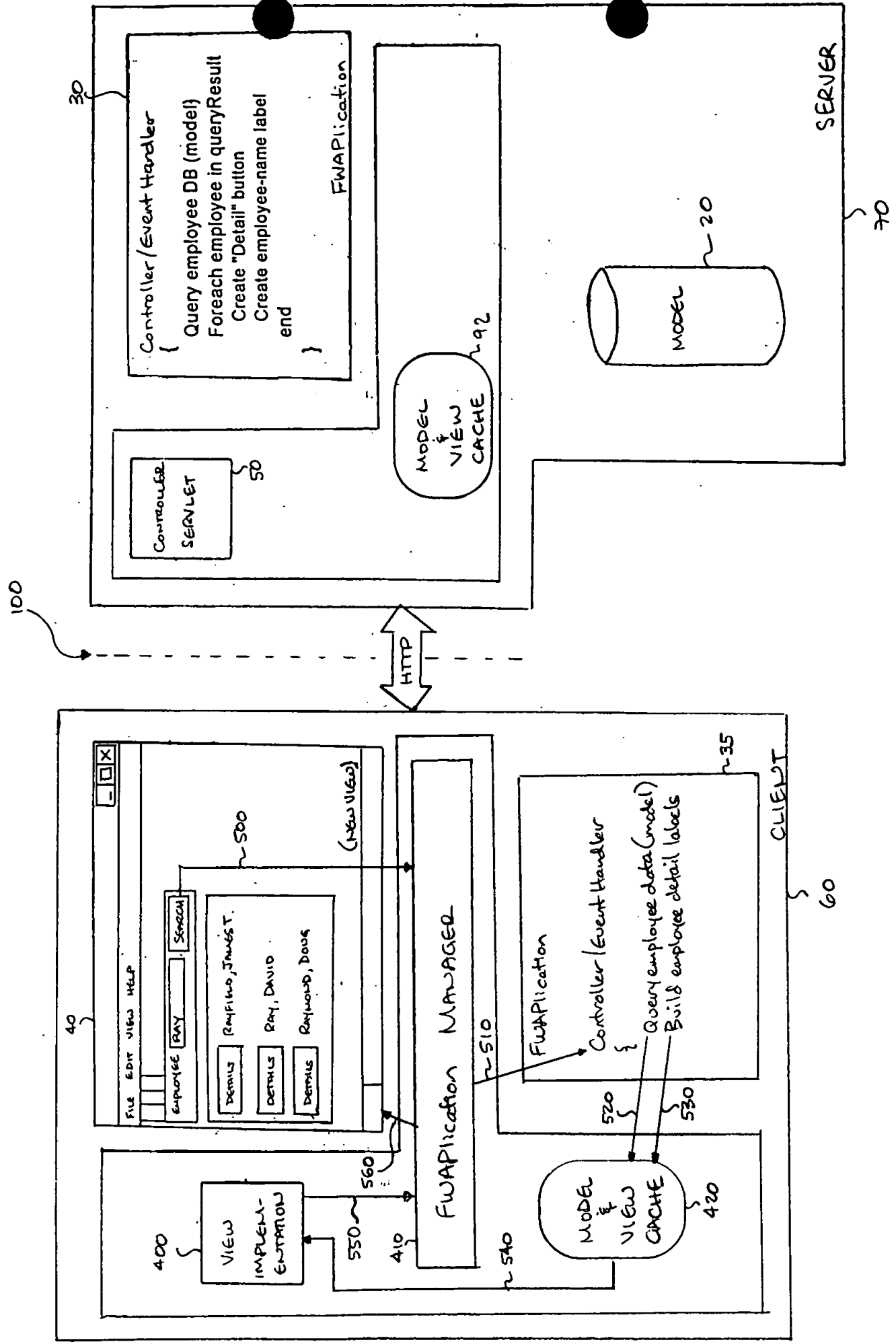
FIG. 4

FIG. 5

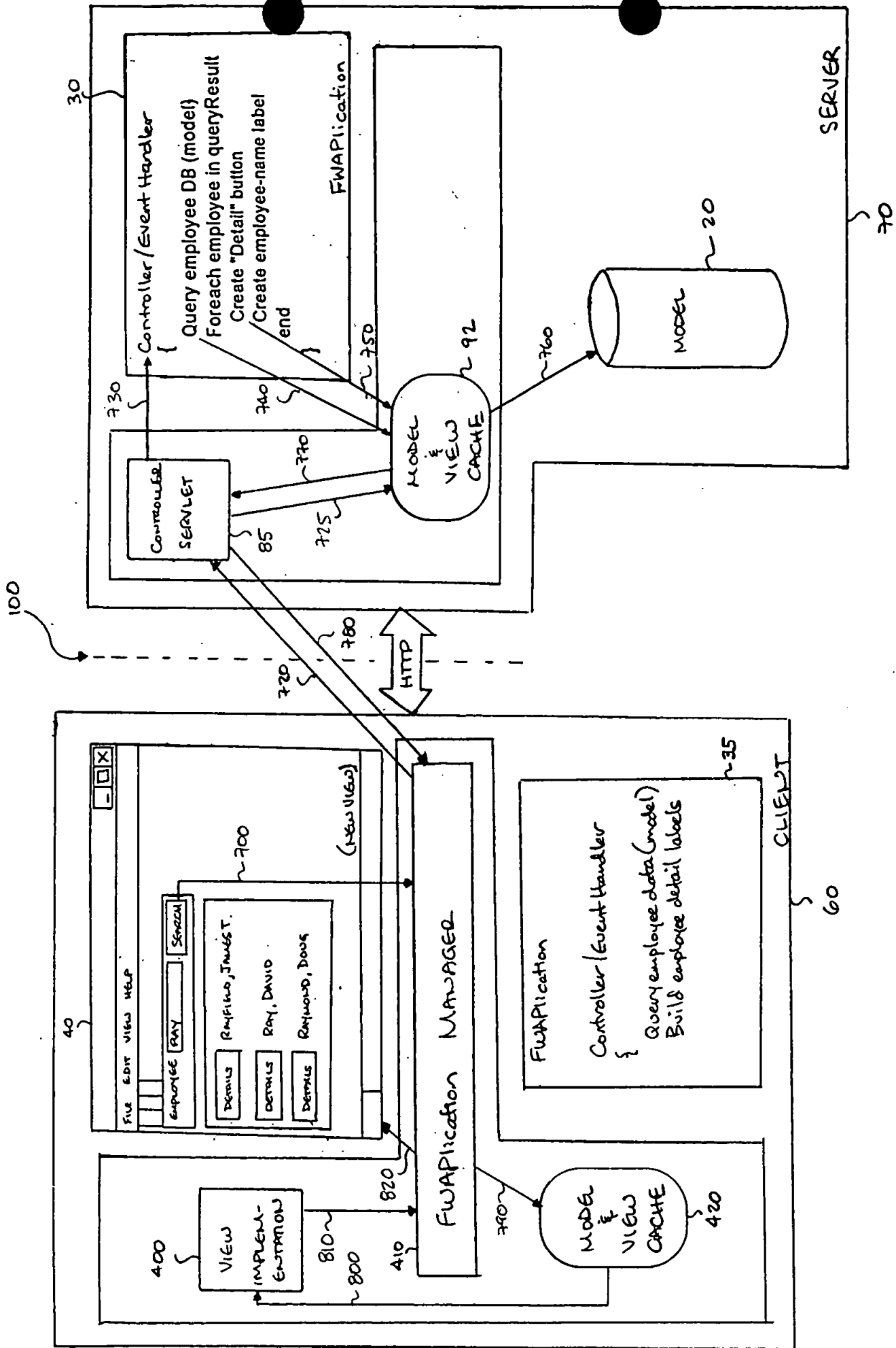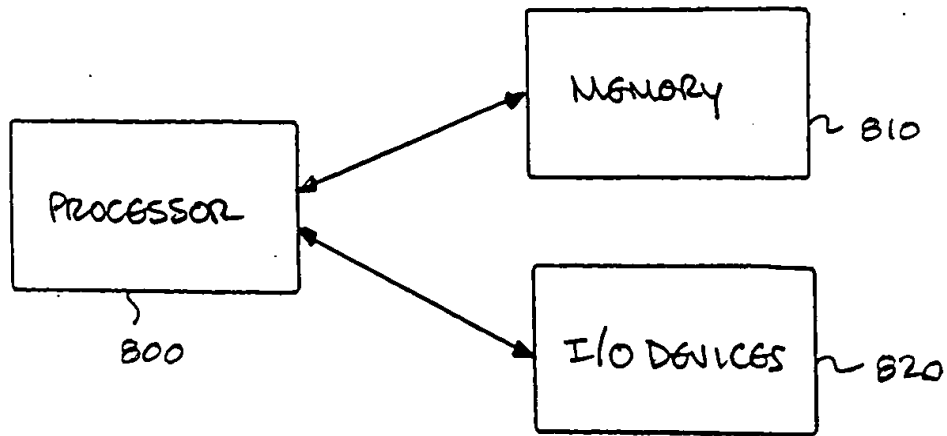FIG. 6

SERVER

30

Controller / Event Handler
{
Query employee DB (model)
Foreach employee in queryResult
 Create "Detail" button
 Create employee-name label
end
}

FWAPlication

750

730

740

770

725

Controller
SERVLET

85

MODEL
&
VIEW
CACHE

92

760

MODEL

20

70

100

780

HTTP

720

File Edit View Help

40

EMPLOYEE   RAY   SEARCH

700

RAYFIELD, JAMES T.
DETAILS

RAY, DAVID
DETAILS

RAYMOND, DOUG
DETAILS

(new view)

FWAPlication MANAGER

410

830

VIEW
IMPLEMEN-
TATION

400

810

800

820

730

MODEL
&
VIEW
CACHE

420

FWAPlication

Controller / Event Handler
{
Query employee data (model)
Build employee detail labels
}

35

60

CLIENT

FIG. 7

FIG. 8